I claim:

1. A method for performance by a coherency controller of a node, comprising:

receiving a transaction relating to a line of local memory of the node; and,

in response to determining locally within the controller that the line of the local memory is one or more of not being cached by any other node, and has not been modified by any other node,

processing the transaction by the coherency controller without accessing information regarding the line of local memory in a tag directory associated with the local memory.

2. The method of claim 1, further comprising, otherwise, processing the transaction by the coherency controller such that the coherency controller accesses information regarding the line of local memory in the tag directory.

3. The method of claim 2, wherein accessing information regarding the line of local memory in the tag directory results in slower processing of the transaction than if information regarding the line of local memory in the tag directory were not accessed.

4. The method of claim 1, wherein determining locally within the controller that the line of the local memory is one or more of not being cached by any other node and has not been modified by any other node comprises examining a table within the coherency controller to determine that the line of the local memory is one or more of not being cached by any other node and has not been modified by any other node.

5. The method of claim 4, wherein looking up the line of local memory within the table of the coherency controller to determine that the line of the local memory is one or more of not being cached by any other node and has not been modified by any other node comprises referencing an entry within the table of the coherency controller corresponding to a section of the local memory including the line of local memory.

6. The method of claim 5, wherein referencing the entry within the table of the coherency controller corresponding to a section of the local memory including the line of local memory comprises looking up within the entry within the table of the controller a count value of a number of lines within the section of local memory being cached by other nodes and a count value of a number of lines within the section of the local memory having been modified by other nodes.

7. The method of claim 5, wherein referencing the entry within the table of the coherency controller corresponding to a section of the local memory including the line of local memory comprises determining from a flag within the entry in the table whether an operating system (OS) has checked out lines within the section of local memory.

8. The method of claim 1, wherein processing the transaction by the coherency controller comprises processing the transaction by a coherency processor of the coherency controller.

9. The method of claim 1, wherein processing the transaction without accessing information regarding the line of memory in the tag directory results in faster processing

of the transaction than if information regarding the line of memory in the tag directory were accessed.

10. A node of a multi-node system comprising:

one or more processors;

local random-access memory (RAM) for the processor(s);

at least one memory controller to process memory operations relative to the local RAM;

at least one coherency controller to process transactions relative to the local RAM; and,

a local region table within each coherency controller storing access information regarding the local RAM by other nodes of the multi-node system so that the coherency controller is able to process the transactions relative to the local RAM without accessing tag directory information associated with the local RAM.

11. The node of claim 10, further comprising a coherency processor within the coherency controller within which the transactions are processed.

12. The node of claim 10, further comprising a tag directory within which the tag directory information is stored.

13. The node of claim 12, further comprising a tag bus connecting the coherency controller to the tag directory.

14. The node of claim 12, wherein the coherency controller accesses the local regional table with less latency than the coherency controller accesses the tag directory.

15. The node of claim 10, wherein the local region table has a plurality of entries corresponding to sections of the local RAM, each entry including a first count value tracking a number of lines of the section of the local RAM being cached by the other nodes and a second count value tracking a number of lines of the section of the local RAM having been modified by the other nodes.

16. The node of claim 10, wherein the local region table has a plurality of flags corresponding to sections of the local RAM, each flag indicating whether another node has checked out lines within the section of the local RAM.

17. The node of claim 10, wherein the coherency controller comprises an application-specific integrated circuit (ASIC).

18. A coherency controller for a node of a system to process transactions relative to at least a portion of memory of the node, comprising:

a coherency processor within which transactions are converted into performable actions to effect the transactions relative to at least the portion of the memory of the node; and,

a local region table storing access information regarding at least the portion of the memory of the node by other nodes of the system so that the coherency controller may be able to convert the transactions within the pipeline without accessing tag directory information associated with the memory of the node.

19. The controller of claim 18, wherein the local region table has a plurality of entries corresponding to sections of at least the portion of the memory of the node, each entry including a first count value tracking a number of lines of the section cached by the other nodes and a second count value tracking a number of lines of the section having been modified by the other nodes.

20. The controller of claim 18, wherein the controller comprises an application-specific integrated circuit (ASIC).